UDC: 519.8 DOI: https://doi.org/10.15407/jai2022.01.245

ARCHITECTURAL AND MATHEMATICAL FUNDAMENTALS OF IMPROVEMENT NEURAL NETWORKS FOR CLASSIFICATION OF IMAGES

V. Slyusar
Central Research Institute of Armaments and Military Equipment of Armed Forces of Ukraine
Povytrophlotsky Ave, 28B, Kyiv, 03049
swadim@ukr.net
https://orcid.org/0000-0002-2912-3149

Abstract. The article proposes ways to solve the problem of structural synthesis of neural networks architectures; based on non-traditional approaches to their analytical formalization and application of new ones data processing operations. The example of the CIFAR10 dataset proves the possibility of improving the accuracy of the solution image classification tasks within an alternative architecture with expandable input and narrowing structures of trained neural networks. The results of the application of input expanding image taken with Resizing and Conv2DTranspose layers at the input trained neural networks indicate the effectiveness of solving such classification problems by example CIFAR10 dataset.

Achieved on the basis of the Xception neural network, the average classification accuracy of 10 classes CIFAR10 images are 97.3%. The side effect of pre-scaling images is leveling the accuracy of the classification of different classes, which allows us to consider such a change in size as an option data augmentation in the dataset. Tensor-matrix methods have been introduced for further development of this approach formalization of the description of neural networks on the basis of the penetrating end product of matrices and its block ones modifications. On this basis, a number of new convolution and maxpooling operations are proposed, as well as combinations on input of the narrowing segment not only symmetrically enlarged image, but also its variants obtained on the basis of a generalized penetrating product. It is about drawing images expanded by rows of pixels (horizontally) and columns (vertically), as well as a combination of several different options symmetrical image extensions on the principle of constructing a pyramidal segment of the PSPNet neural network.

Keywords: neural network, tensor-matrix theory, penetrated face-splitting product of matrices.

АРХІТЕКТУРНО-МАТЕМАТИЧНІ ОСНОВИ УДОСКОНАЛЕННЯ НЕЙРОННИХ МЕРЕЖ З КЛАСИФІКАЦІЇ ЗОБРАЖЕНЬ

В. І. Слюсар
Центральний науково-дослідний інститут озброєння та військової техніки Збройних Сил України, Україна
пр. Повітрофлотський, 28Б, м. Київ, 03049
swadim@ukr.net
https://orcid.org/0000-0002-2912-3149

Анотація. В статті запропоновані шляхи вирішення завдання структурного синтезу нейромережних архітектур, спираючись на нетрадиційні підходи щодо їх аналітичної формалізації та застосування нових операцій обробки даних. На прикладі датасету CIFAR10 доведено можливість підвищення точності вирішення завдань класифікації зображень в рамках альтернативної архітектури з розширювальною вхідною та звужуючою структурами навчених нейронних мереж. Результати застосування укрупнення вхідного зображення, що було здійснене за допомогою прошарків Resizing та Conv2DTranspose на вході попередньо навчених нейромереж, свідчать про ефективність вирішення у такий спосіб завдань класифікації на прикладі датасету CIFAR10. Досягнута на основі нейромережі Xception середня по 10-ом класам точність класифікації зображень CIFAR10 становить 97,3%. Супутній ефект попереднього масштабування зображень полягає у вирівнюванні точності класифікації різних класів, що дозволяє розглядати таку зміну розмірів як варіант аргументації даних у датасеті. Для подальшого розвитку цього підходу запроваджено тензорно-матричні методи формалізації опису нейронних мереж на основі проникаючого торцевого добутку матриць та його блокових модифікацій. На цій базі запропоновано низку нових операцій згортки та макспулінгу, а також поєднання на вході звужувального сегменту не тільки симетрично збільшеного зображення, а й його варіантів, отриманих на основі узагальненого проникаючого добутку. Мова йде про залучення зображень, розширених по рядках пікселів (по горизонталі) та по стовпцях (по вертикалі), а також поєднання кількох різних варіантів симетричних розширень зображень за принципом побудови пірамідального сегменту нейромережі PSPNet.

Ключові слова: нейронна мережа, тензорно-матрична теорія, проникаючий торцевий добуток матриць.

Вступ

Необхідність міграції штучного інтелекту у пристрої користувачів вимагає переходу до нових, потужніших обчислювальних схем, розробкою яких сьогодні займаються багато провідних компаній світу. Йдеться, наприклад, про використання тензорних обчислювальних пристроїв, більш потужної процесорної техніки та графічних прискорювачів, а також новації в архітектурі нейронних мереж. Разом з тим, іншим стримуючим чинником є застаріла, традиційна математична концепція, коли для обробки даних використовуються звичайні матричні операції. Цілком очевидно, що розвиток архітектурних рішень в технологіях нейромереж має спиратися й на оновлення відповідного математичного апарату. Така потреба робить актуальним завдання структурного синтезу нейромережних архітектур, спираючись на нетрадиційні підходи щодо їх аналітичної формалізації та застосування нових операцій обробки даних.

Аналіз останніх досліджень і публікацій

Просування у зазначеній сфері буде суттєвим, якщо запровадити в бібліотеці програмування нейронних мереж та їх навчання нові підходи, у тому числі розвинений автором тензорно-матричний апарат, одним з компонентів якого є торцевий добуток матриць [1-5].

Як відомо, торцевий добуток зводиться до множення кожного елемента лівої матриці на весь рядок правої, відповідно до порядкового номера цього елемента [1-5]. Альтернативою даної операції є стовпцевий добуток Хатрі-Рао, при якому елемент лівої матриці множиться на відповідний за номером стовпець правої. У [1-5] представлені властивості цих матричних операцій, їх комбінацій між собою та з кронекерівським добутком. Особливо важливою є можливість переходу до адамарового добутку, що спрощує обчислювальні операції. Серед інших властивостей інтерес представляє, зокрема, еквівалентність даних операцій та добутку Кронекера щодо векторів, а також теорема, що зв'язує торцевий та стовпцевий добутки.

Як відомо, операція торцевого добутку використовується в рамках тензорного скетчу [6], де вона дозволяє замінити згортку від перетворення Фур'є матрично-векторного добутку еквівалентним добутком Адамара відповідних Фур'є-згорток матриць. Це істотно знижує кількість операцій у тому випадку, якщо нейронна мережа трансформує вихідний простір великих даних у простір меншої розмірності, що, як правило, спостерігається у згорткових нейромережах.

Наступним узагальненням зазначених матричних операцій є їх блокові варіанти, коли відповідні процедури виконуються поблочно [3]. Наприклад, кожна пара блоків 2-х матриць з однаковим розбиттям на блоки множиться за принципом торцевого добутку [3]. Аналогічна операція можлива також для блочного стовпцевого добутку Хатрі-Рао.

Відносно блокових матриць виконується та ж властивість транспонування добутків і складніша фундаментальна властивість перемноження блоків матриць з урахуванням їх блочно-стовпцевої та блочно-транспонованої структури.

Необхідність введення блокових операцій торцевого та стовпцевого (Хатрі-Рао) добутків зумовлена тим, що при диференціюванні, наприклад, стовпцевого добутку матриць, перехід до блокового варіанту дозволяє факторизувати операції диференціювання та знаходження градієнтів, що дуже важливо при описі процедур навчання нейронних мереж.

Як відомо, метод зворотного спуску реалізується на основі обчислення градієнта, при цьому зазначені блокові операції демонструють можливість спрощення такої процедури та її автоматизації. На цій основі може бути сформована й нижня межа Крамера-Рао [5], що дозволяє визначити потенційну

ефективність нейронних мереж в умовах впливу шумів. Зокрема, з даною межею треба порівнювати можливості штучного інтелекту при вирішенні завдання оцінювання параметрів сигналів. Хоча у деяких випадках виявляється, що нейронні мережі дають кращу оцінку точності, ніж нижня межа Крамера-Рао, це питання потребує детального дослідження. Наразі подібні факти пояснюються тим, що нейронна мережа враховує більше чинників, ніж та модель, у межах якої описується аналізований процес.

У разі впливу нейронних мереж на розвиток відомого тензорно-матричного апарату, автором було запропоновано проникаючий торцевий добуток матриць [3]. Згідно з визначенням, введеним в [3], для p×g-матриці A та n-вимірного тензору B, розгорнутого в блокову матрицю рядків чи стовпців $B=[B_n]$ ($n > 1$) з p×g-блоками, їх проникаючий торцевий добуток має вид:

$$A \blacksquare B = [A \circ B_r], \tag{1}$$

де $A \circ B_r$ - добуток Адамара.

Якщо тензор B є блок-рядком, отримаємо:

$$A \blacksquare B=[A \circ B_r]=[A \circ B_1 \vdots \ldots \vdots A \circ B_r \vdots \ldots].$$

Наприклад:

$$A=\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \\ a_{31} & a_{32} \end{bmatrix},\ B=\left[\begin{array}{cc:cc:cc} b_{111} & b_{121} & b_{112} & b_{122} & b_{113} & b_{123} \\ b_{211} & b_{221} & b_{212} & b_{222} & b_{213} & b_{223} \\ b_{311} & b_{321} & b_{312} & b_{322} & b_{313} & b_{323} \end{array}\right],$$

$$A \blacksquare B =$$

$$=\left[\begin{array}{cc:cc:cc} a_{11}\cdot b_{111} & a_{12}\cdot b_{121} & a_{11}\cdot b_{112} & a_{12}\cdot b_{122} & a_{11}\cdot b_{113} & a_{12}\cdot b_{123} \\ a_{21}\cdot b_{211} & a_{22}\cdot b_{221} & a_{21}\cdot b_{212} & a_{22}\cdot b_{222} & a_{21}\cdot b_{213} & a_{22}\cdot b_{223} \\ a_{31}\cdot b_{311} & a_{32}\cdot b_{321} & a_{31}\cdot b_{312} & a_{32}\cdot b_{322} & a_{31}\cdot b_{313} & a_{32}\cdot b_{323} \end{array}\right].$$

У цьому випадку, як показано в [3], виконується перестановочна властивість як щодо добутків матриць, так і щодо векторів, що застосовуються в нейронних мережах.

Слід зазначити, що в бібліотеці TensorFlow [7] вже реалізований спрощений варіант проникаючого добутку, який полягає в тому, що вектор множиться на матрицю повністю, або матриця множиться на вектор [8]. У цьому випадку кожен елемент вектора методом добутку Адамара перемножується на відповідний стовпець матриці. Аналогічний підхід був розглянутий також в [9], де описана модель нейронної гіпермережі на основі операції поелементного добутку вектора на матрицю, що є спрощеним варіантом проникаючого торцевого добутку. Однак вказана бібліотека TensorFlow не працює у разі перемноження матриць та блокових матриць, тому запропонований розвиток тензорно-матричної теорії є досить корисним, щоб прискорити реалізацію обчислень під час навчання нейромереж.

Розглянутий тензорно-матричний апарат дозволяє нарощувати арсенал методів, що використовуються при описі нейромереж, оскільки проникаючий добуток та його блокові версії дають вихідні масиви зважених даних або пікселів, з якими далі можна виконувати довільні операції. Однак зазначені напрацювання потребують подальшого розвитку у поєднанні з новими архітектурними рішеннями, чому й присвячена дана стаття.

Постановка проблеми

Типові згорткові нейронні мережі, що вирішують завдання класифікації, мають архітектуру, яка звужується від входу нейромережі до її виходу зі зменшенням розмірності тензору даних. Прикладом тому є структури нейронних мереж LeNet5 [10, 11], AlexNet [12, 13], VGG16 [14], DenseNet [15], MobileNet [16 - 18], NASNet [19] та ін. У той же час нейромережі U-Net [20], генеративно-змагальні мережі (GAN) [21] та ін. мають у своєму складі й сегменти архітектури, що розширюється у напрямку виходу. Як засвідчили проведені в [22] дослідження, інтеграція в згорткові мережі для завдань класифікації фрагментів архітектури, що розширюється, може позитивно вплинути на їх точність. Разом з тим, таке удосконалення архітектурних рішень має спиратися на відповідний розвиток тензорно-матричного апарату нейромереж. Це дозволить збагатити можливості

розробників більш широким асортиментом операцій з обробки даних та надасть додаткову варіативність нейронній схемотехніці.

Мета дослідження

Метою роботи є дослідження запропонованої автором архітектури нейромереж для класифікації зображень на основі чередування ланок розширення та звуження архітектури на прикладі використання датасету CIFAR10, а також розвиток відповідного тензорно-матричного апарату обробки даних.

Виклад основного матеріалу

Метод підвищення точності класифікації нейромереж

Перш ніж перейти до опису отриманих результатів, слід вказати, що вибір для тестування нейромереж датасету CIFAR10 [23] був зумовлений його відносною простотою. Зокрема, інший відомий датасет зображень CIFAR 100 [23] є дуже громіздким і потребує, наприклад, для виконання 20 епох навчання нейромережі Xcepttion в Google Colab Pro+ з використанням GPU близько 6 годин 28 хв. На основі ж CIFAR10 така нейромережа навчається приблизно в 10 разів швидше.

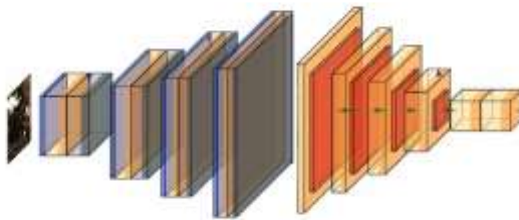


Рис. 1. Пропонована архітектура згорткової мережі для класифікації

Тестування відповідних нейронних структур здійснювалося у фреймворку “Terra AI”, що на даний момент є найкращим у своєму класі засобом синтезу та тестування нейромереж з точки зору досягнутого ступеня автоматизації та зручності використання.

Особливістю іншої з таких архітектур є послідовне включення попередньої навченої на датасеті ImageNet [12, 24] нейромережі VGG16 і LSTM (рис. 2). При цьому для узгодження розмірностей масиву даних на виході VGG16 (вектор з 512 елементів) з входом LSTM були використані два послідовно включені прошарки Reshape і Conv1DTranspose. Шар Reshape формував з 512-елементного вектора матрицю формату 16×32, яка трансформувалася далі за допомогою лінійки Conv1DTranspose в матрицю 16×128. Параметри LSTM-шару забезпечували на виході вектор даних 256 елементів. Досягнута в результаті середньокласова точність характеризується величиною 88,41%.

Цей результат можна порівняти з точністю, забезпеченою попередньо навченою мережею VGG16, в якості вихідних ланцюгів якої використані прошарки Dense з характерною для LeNet5 розмірністю (120 і 84 елементи), що відрізняються заміною функції активації на ReLu. Стартувавши з 80,39% середньокласової точності, така нейронна мережа забезпечила на 15-ій епосі навчання точність 88,57%. Аналогічна структура на основі DenseNet201, у разі постійного кроку навчання 0,00001, призвела до точності 88,3% на 18-ій епосі.

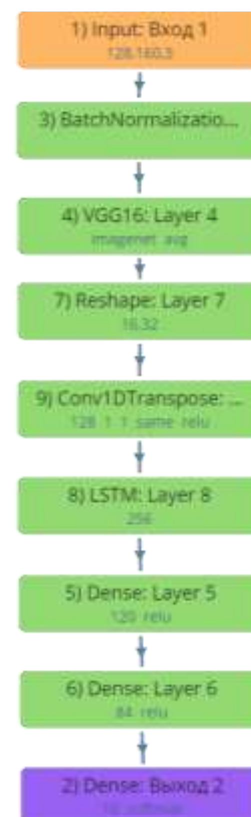


Рис. 2. Послідовна архітектура з використанням попередньо навченої нейромережі

Таким чином, отримані результати дозволяють зробити висновок, що на зображеннях малих розмірів майже всі топові нейронні мережі працюють приблизно однаково. При цьому більш високу точність дають нейронні мережі, що забезпечують найкращі результати на датасеті ImageNet.

Разом з тим, винятком із цього ряду стала інтеграція в архітектуру рис. 1 попередньо навченої мережі Xception. Щоб узгодити розмірність фотографій CIFAR10 (32×32×3) з необхідним форматом зображень на вході Xception (229×229×3 пікселів), вихід шару нормалізації був заведений на додатково включений шар Resizing в білінійному режиму. У результаті така структура нейронного класифікатора дала різкий стрибок в середній точності до 97,2% вже на 9-ій епосі. Відповідна матриця точності представлена на рис. 3. Надалі цей показник був збільшений до 97,3%.

Аналогічний ефект значного приросту в середній точності (95,6%) був отриманий і у разі використання замість Xception навченої мережі InceptionV3 у тій самій схемі включення (рис. 4).

У якості гіпотези щодо можливої причини такого зростання показників точності слід розглядати вплив штучного укрупнення вхідного зображення, що було здійснене за допомогою шару Resizing. Прояв такого ефекту на прикладі двох нейромереж (Xception і InceptionV3) дає підстави вважати, що він є досить загальним і може бути свідченням певної закономірності, характерної для класифікаційних нейромереж. Подальші дослідження з перевірки цього припущення довели його справедливість. Зокрема, у випадку NASNetMobile застосування на вході нейромережі Resizing до формату 224 х 224 (рис. 5) дозволило отримати усереднену точність класифікації зображень 97%.

	airplane	autom...	bird	cat	deer	dog	frog	horse	ship	truck
airplane	974 97.4%	0 0%	5 0.5%	3 0.3%	2 0.2%	0 0%	0 0%	1 0.1%	12 1.2%	3 0.3%
automobile	0 0%	985 98.5%	0 0%	0 0%	0 0%	0 0%	0 0%	0 0%	4 0.4%	11 1.1%
bird	6 0.6%	0 0%	963 96.3%	12 1.2%	8 0.8%	7 0.7%	3 0.3%	1 0.1%	0 0%	0 0%
cat	2 0.2%	0 0%	4 0.4%	927 92.7%	7 0.7%	47 4.7%	5 0.5%	3 0.3%	2 0.2%	3 0.3%
deer	1 0.1%	0 0%	2 0.2%	9 0.9%	978 97.8%	2 0.2%	1 0.1%	7 0.7%	0 0%	0 0%
dog	0 0%	0 0%	4 0.4%	35 3.5%	11 1.1%	946 94.6%	1 0.1%	3 0.3%	0 0%	0 0%
frog	0 0%	0 0%	6 0.6%	4 0.4%	0 0%	3 0.3%	987 98.7%	0 0%	0 0%	0 0%
horse	0 0%	0 0%	3 0.3%	4 0.4%	12 1.2%	4 0.4%	0 0%	977 97.7%	0 0%	0 0%
ship	8 0.8%	2 0.2%	1 0.1%	0 0%	0 0%	0 0%	2 0.2%	0 0%	985 98.5%	2 0.2%
truck	4 0.4%	20 2%	0 0%	0 0%	0 0%	0 0%	0 0%	0 0%	3 0.3%	973 97.3%

Рис. 3. Матриця точності нейромережі на основі Xception

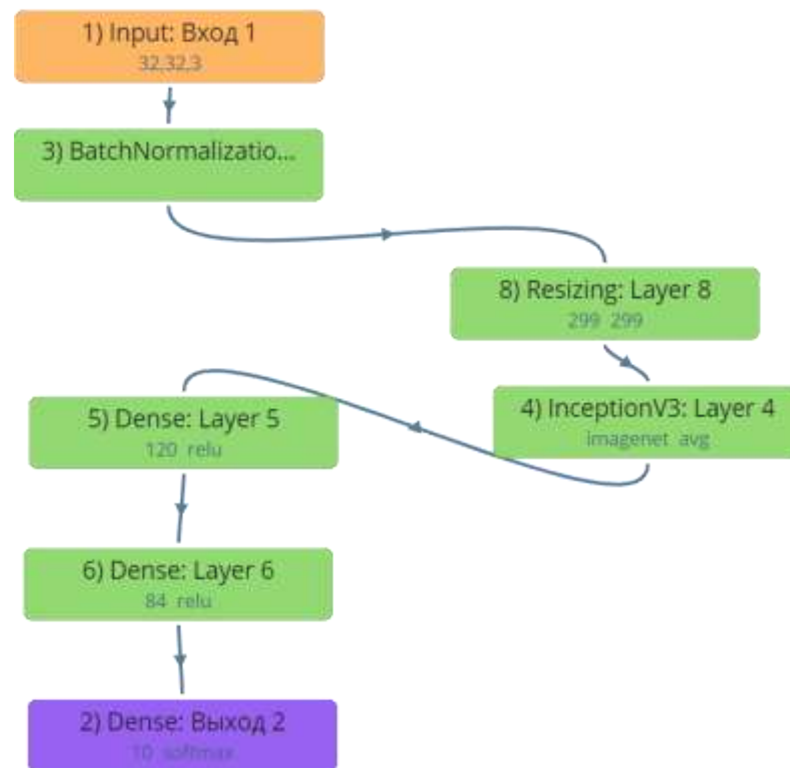


Рис. 4. Схема підключення навченої нейромережі InceptionV3

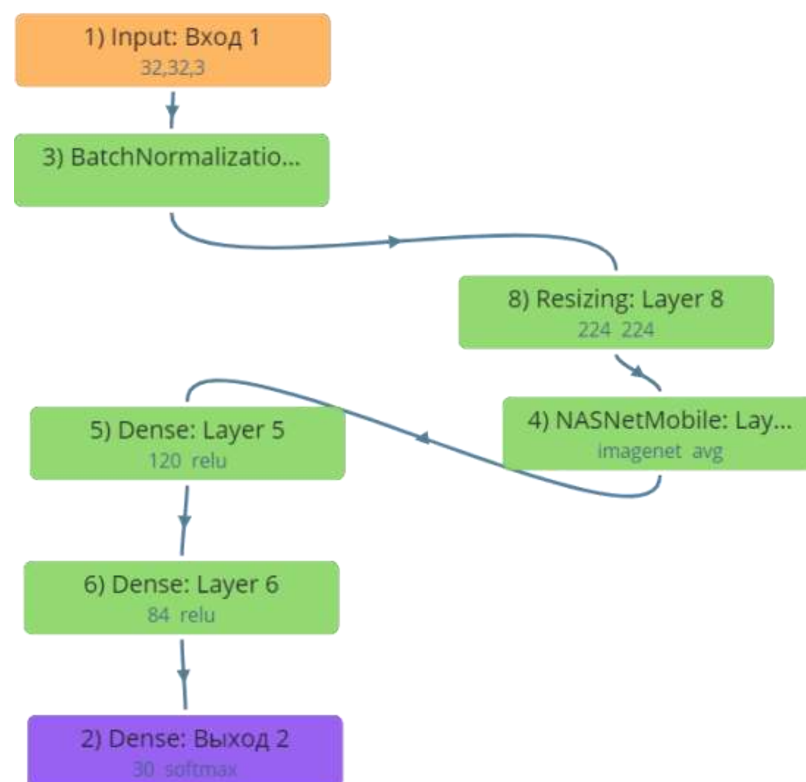


Рис. 5. Схема підключення навченої нейромережі NASNetMobile

Цей же підхід для VGG16 із Resizing до формату 331 x 331 призвів до зростання точності до величини 91,1%. У межах аналогічної архітектури попередньо навчена нейромережа DenseNet201 забезпечила 95,5% точності на 6-ій епосі, а потім на 44-ій – 97,1%. При цьому Resizing був зроблений до формату 224x224, оскільки при намаганні використати формат 331x331 необхідні обчислювальні витрати перевищили ресурси, надані на той момент Google Colab Pro+.

Таким чином, ефективність методу залежить від структури нейромережі, але загалом він підвищує точність.

Слід зазначити, що в якості прошарку, що формує збільшене зображення на вході навченої нейромережі, може бути використаний Conv2Dtranspose (рис. 6). В основі його лежить застосування кронекерівського добутку матриць. Зокрема, стосовно датасету CIFAR10 (32×32×3) вхідні зображення можуть бути збільшені в 7 разів, до формату (224×224×3) шляхом вибору наступних параметрів прошарку Conv2DTranspose: кількість фільтрів – 3, розмір ядра згортки – 7x7, розмір кроку зсуву ядра – 7x7.

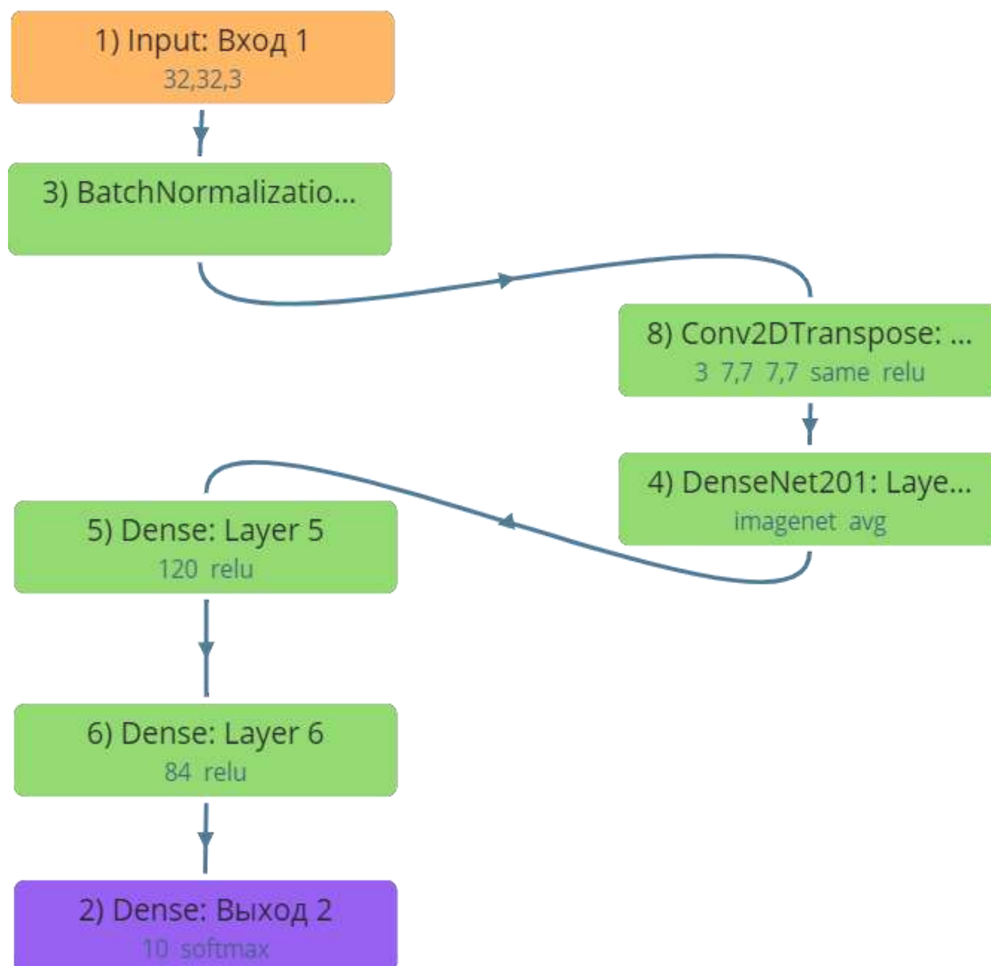


Рис. 6. Архітектура мережі з прошарком Conv2DTranspose

Таке рішення у випадку тієї ж DenseNet201 дещо програло в точності застосуванню прошарку Resizing, але, при старті з початкових 90,4%, дозволило отримати точність класифікації до 96% на 49-ій епосі навчання.

Аналогічні дослідження було проведено для DenseNet169. У випадку схеми із шаром Conv2DTranspose на 10-ій епосі була зафіксована точність 89,5%. Далі з переходом від кроку навчання 0,001 до 0,0001 точність підвищилася з 93,3% до 93,6% на 10-ій епосі. Як і раніше, найгірше за точністю класифікуються класи cat, dog i bird.

Однією з причин досягнення помітного приросту точності є більш чітка сепарація класів зображень, коли відмінності у вмісті їх ключових ознак багаторазово збільшуються на площі аналізованого знімка.

Крім того, суттєвий супутній ефект навмисного масштабування зображень полягає у вирівнюванні точності класифікації різних класів, оскільки на великих форматах наявний дисбаланс якості зображення вихідних знімків різних класів нівелюється. Зазначений ефект дозволяє розглядати таку зміну розмірів як варіант аргументації даних у датасеті.

Разом з тим очевидно, що для масштабування вхідних зображень краще застосовувати більш складні процедури, які дозволяють підвищити різкість збільшених зображень. Фактично на вході класифікуючої нейромережі повинна стояти спеціальна нейромережа оптимального масштабування вихідного зображення. Як варіант, у цьому випадку доцільно використати GAN, яка збільшує зображення і поліпшує його чіткість [25]. Крім того, для масштабування зображень та їх подальшої обробки доцільно застосувати розвинений нижче тензорно-матричний апарат формалізації функціонування нейронних мереж.

Розвиток тензорно-матричного апарату нейромереж для завдань класифікації

Завдання, винесене у заголовок цього розділу, може бути вирішене, спираючись на вже згадану вище концепцію проникаючого торцевого добутку матриць (1) [3]. Щоб наочно уявити його суть у контексті класифікації зображень, можна домовитися, що матриця A, наприклад, є вхідним зображенням, а блок-матриця B - це набір вагових коефіцієнтів різних нейронів, наприклад, першого шару. Відповідно, у цьому випадку кожен елемент або піксель зображення множиться на ваговий нейронний коефіцієнт.

Для того, щоб формалізувати згорткові нейронні мережі, відповідний результат проникаючого добутку необхідно помножити на вектор одиниць праворуч $((A ◙ B)\times 1)$, або на вектор-рядок зліва $(1^T(A ◙ B))$, або на блоковий вектор одиниць праворуч за процедурою блокового традиційного множення:

$$(A ◙ B)\ [\times] 1, \qquad (2)$$

де [×] – символ блокового звичайного добутку матриць, 1 – блок-вектор одиниць.

Щоб отримати скаляр, необхідно таке множення здійснити зліва та справа:

$$1^T(A ◙ B)1. \qquad (3)$$

При такому підході опис нейрона зводиться до функції активації з аргументом у вигляді проникаючого добутку, помноженого на вектор одиниць, що еквівалентно процедурі згортки, наприклад:

$$\tanh[(A ◙ B)\times 1+d], \qquad (3)$$

або

$$SReLU[1^T(A ◙ B)1+d], \qquad (4)$$

де d - вектор (скаляр).

Такий запис дозволяє компактно формалізувати складну архітектуру нейронних мереж, що складається з кількох шарів.

Даний підхід забезпечує поєднання традиційної згортки з попередньою процедурою перетворення Фур'є або вейвлетного перетворення. Це дозволить виділити додаткові ознаки, особливо якщо ядро згортки більше 2-х, наприклад, 4, 6 і

т.д. Аналогічно, зазначена процедура може комбінуватися з операцією maxPooling. На даний момент відомі два її варіанти: вибір максимальних значень або формування середньозваженої суми по всьому блоку. Пропонується попередньо перед цим виконувати також перетворення Фур'є або вейвлет-перетворення і далі вибирати максимальний відгук. Це дозволить певною мірою усереднити шум у заданій області та якось спрогнозувати спектральний склад фрагмента даних. Аналогічне узагальнення можливе на випадок 3-вимірних вхідних масивів, коли використовується, наприклад, RGB-матриця, або багатовимірний тензор, що поєднує множину даних з кількох відеокамер у різних спектральних діапазонах. У цьому випадку в нейромережі застосовуються 3-вимірні або багатовимірні згортки, причому 3-вимірна згортка вже використовується в даний час, проте, комбінація її з перетворенням Фур'є, вейвлетами або іншими функціями поки не застосовувалася.

Описаний підхід може бути узагальнений на різні варіанти комбінацій проникаючого добутку та підсумовування всіх елементів матричного блоку, виконання макспулінгу тощо. Можливо, наприклад, ввести проміжну операцію, коли сума формується лише по рядках блоку, і далі макспулінг застосовується в межах кожного з рядків, внаслідок чого формується векторний відгук, або ж аналогічна сума формується по стовпцях. Можливо застосувати макспулінг після перетворення Фур'є і т.д. Власне, йдеться про "зоопарк" згорток для нейромереж за аналогією з назвою Інтернет-сайту "Зоопарк нейромереж" [26].

Формалізований запис деяких з пропонованих операцій на основі перетворення Фур'є наведено в табл. 1. Аналогічні операції синтезуються на основі вейвлет-перетворень. В принципі, кожну таку операцію можна позначати своїм математичним символом або вводити спеціальні найменування функцій, що словесно описують суть маніпуляцій з даними.

Таблиця 1."Зоопарк" згорток

Позначка операції	Сутність операції	
$\boldsymbol{A}_1 \text{◙}_{\Sigma} \boldsymbol{B}_1$	Проникаючий торцевий добуток (ПТД) з подальшим підсумовуванням усіх елементів отриманої матриці	
$\boldsymbol{A}_1 \text{◙}_{M} \boldsymbol{B}_1$	ПТД з подальшим відбором максимального елемента серед усіх елементів отриманої матриці (операція MaxPull)	
$\boldsymbol{A}_1 \text{◙}_{F} \boldsymbol{B}_1$	ПТД з подальшим швидким перетворенням Фур'є (ШПФ) кожного з рядків отриманої матриці	
$\boldsymbol{A}_1 \text{◙}_{FT} \boldsymbol{B}_1$	ПТД з подальшим ШПФ кожного з стовпців отриманої матриці	
$\boldsymbol{A}_1 \text{◙}_{2F} \boldsymbol{B}_1$	ПТД з подальшим 2-вимірним ШПФ кожного з блоків отриманої матриці	
$\boldsymbol{A}_1 \text{◙}_{3F} \boldsymbol{B}_1$	ПТД з подальшим 3-вимірним ШПФ отриманої матриці	
$\boldsymbol{A}_1 \text{◙}_{KF} \boldsymbol{B}_1$	ПТД з подальшим K-вимірним ШПФ отриманої матриці	
$\boldsymbol{A}_1 \text{◙}_{\Sigma M} \boldsymbol{B}_1$	ПТД з подальшим підсумовуванням елементів у кожному з рядків отриманої матриці та відбором максимального елемента у вихідному векторі (операція MaxPull)	
$\boldsymbol{A}_1 \text{◙}_{\Sigma TM} \boldsymbol{B}_1$	ПТД з подальшим підсумовуванням елементів у кожному стовпці отриманої матриці та відбором максимального елемента у вихідному вектор-рядку (операція MaxPull)	
$\boldsymbol{A}_1 \text{◙}_{F\Sigma} \boldsymbol{B}_1$	ПТД з подальшим ШПФ кожного з рядків отриманої матриці та формуванням вихідного рядка шляхом міжрядкової суми елементів з однаковими номерами	
$\boldsymbol{A}_1 \text{◙}_{FM} \boldsymbol{B}_1$	ПТД з подальшим ШПФ кожного з рядків отриманої матриці та формуванням вихідного вектора шляхом відбору максимального елемента в отриманих рядках (операція MaxPull)	
$\boldsymbol{A}_1 \text{◙}_{F\Sigma M} \boldsymbol{B}_1$	ПТД з подальшим ШПФ кожного з рядків отриманої матриці, формуванням вихідного рядка шляхом міжрядкової суми елементів з однаковими номерами та відбором максимального	

	елемента в отриманому рядку (операція MaxPull)	
$\boldsymbol{A}_1$ ◙$_{2FM}$ $\boldsymbol{B}_1$ $\boldsymbol{A}_1$ ◙$_{3FM}$ $\boldsymbol{B}_1$ $\boldsymbol{A}_1$ ◙$_{KFM}$ $\boldsymbol{B}_1$	ПТД з подальшим 2(3)(K)-вимірним ШПФ кожного з блоків отриманої матриці та відбором максимального з елементів (MaxPull)	
$\boldsymbol{A}_1$ ◙$_{FT\sum}$ $\boldsymbol{B}_1$	ПТД з подальшим ШПФ кожного з стовпців отриманої матриці та формуванням вихідного вектора шляхом міжстовпцевої суми елементів з однаковими номерами	
$\boldsymbol{A}_1$ ◙$_{FTM}$ $\boldsymbol{B}_1$	ПТД з подальшим ШПФ кожного з стовпців отриманої матриці та подальшим формуванням вихідного рядка шляхом відбору максимального елемента в отриманих стовпцях	
$\boldsymbol{A}_1$ ◙$_{FT\sum M}$ $\boldsymbol{B}_1$	ПТД з подальшим ШПФ кожного з стовпців отриманої матриці, формуванням вихідного вектора шляхом міжстовпцевої суми елементів з однаковими номерами та відбором максимального елемента в отриманому векторі (операція MaxPull)	

Іншим типом добутку, що розширює можливості роботи з нейромережами, є проникаючий кронекерівський добуток, суть якого полягає в тому, що кожен блок лівої матриці множиться за принципом прямого добутку на всю матрицю праворуч за допомогою проникаючого добутку. Зокрема, для тензорів A і B, розгорнутих у блокові матриці з блоками однакової розмірності, відповідна операція має запис:

$$A\ [◙]\ B=[A_{ij}\ ◙\ B]=[A_{ij}\circ B_{mr}]. \quad (5)$$

Ця процедура корисна при математичній формалізації обробки в нейромережі відеопотоку або множини кадрів з різних відеокамер одночасно. При цьому кожен кадр зображення зважується на свою вагову матрицю відповідно до множини нейронів, у тому числі кількох шарів. За допомогою даної операції було запропоновано аналітичний опис послідовності виконання обробки даних у різних прошарках моделей LeNet5 [11] та AlexNet [13]. Суттєво, що у поєднанні з проникаючим кронекерівським добутком теж можна застосовувати різні варіанти згорток, аналогічно наведеним в табл. 1, з акцентом на їх 3-вимірних та багатовимірних версіях.

Що стосується більш складних мереж, в яких мають місце не тільки архітектура, що звужується, але й гілки, що розширюються (наприклад GAN), то в цьому випадку розглянутий матричний апарат є недостатнім і може бути доповнений процедурою узагальненого проникаючого добутку.

Суть його полягає у тому, що кожен блок лівої матриці множиться за принципом проникаючого торцевого добутку на відповідний рядок блоків правої матриці. Відповідно, синтез архітектури нейромережі, що розширюється, обмежується тільки рядками блоків. У даний момент для переходу до структур, що розширюються, застосовується тільки прямий добуток матриць, тобто один елемент множиться на якусь вагову матрицю формату 2x2, 3x3, 4x4 і т.д. (див. Conv2Dtranspose). Хоча це дозволяє поступово збільшити розмірність і перейти від малої розмірності на вході до збільшеної на виході нейронної мережі, однак у пропонованому підході застосовується більш гнучкий апарат. У його рамках можна робити розширення тільки по рядках або як альтернатива - по стовпцях блоків, що забезпечується узагальненим транспонованим добутком.

У більш широкому випадку, така процедура може виконуватися поблоково, аналогічний підхід може застосовуватися і до блочного узагальненого проникаючого добутку. Такі тензорно-матричні процедури дозволяють нарощувати структури нейромереж, застосовуючи їх ансамблі, кластери або конгломерати різної ієрархії паралельно функціонуючих нейросегментів, зокрема фрактальної структури [27].

У контексті розглянутої у першому розділі архітектури з розширенням та звуженням структури мережі (рис. 1) заслуговує на увагу поєднання на вході звужуваного сегменту не тільки симетрично збільшеного зображення, а й його варіантів, отриманих на основі узагальненого проникаючого добутку. Мова йде про залучення зображень,

розширених по рядках пікселів (по горизонталі) та по стовпцях (по вертикалі) (рис. 7).

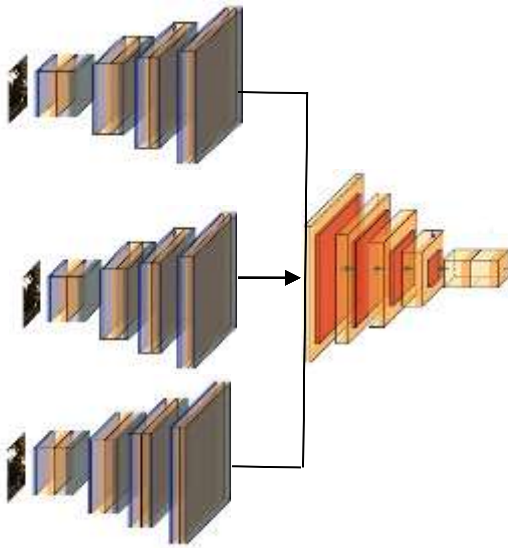


Рис. 7. Архітектура згорткової мережі з поєднанням різних типів розширення зображень

Крім того, можливе поєднання кількох різних варіантів симетричних розширень зображень за принципом побудови пірамідального сегменту нейромережі PSPNet [28].

Аналогічне узагальнення архітектури пропонується використовувати і на основі проникаючого кронекерівського добутку у блочному варіанті реалізації, коли ця процедура виконується поблоково.

На закінчення, слід зазначити, що запропонована різноманітність згорткових операцій у поєднанні з розвиненим тензорно-матричним апаратом надасть додаткові можливості розробникам щодо освоєння нових концепцій нейромереж та їх практичної реалізації.

Важливою умовою, як вже зазначалося, є інтеграція запропонованих тензорно-матричних операцій у бібліотеки фреймворків проєктування та навчання нейронних мереж.

Висновки

За отриманими у роботі результатами можна зробити наступні висновки:

- в нейромережах класифікації зображень для підвищення точності доцільно використовувати архітектуру з вхідною ланкою розширення зображень;
- проведені результати застосування укрупнення вхідного зображення, що було здійснене за допомогою прошарків Resizing та Conv2DTranspose на вході попередньо навчених нейромереж, свідчать про ефективність вирішення у такий спосіб завдань класифікації на прикладі датасету CIFAR10;
- суттєвий супутній ефект масштабування зображень шляхом їх збільшення полягає у вирівнюванні точності класифікації різних класів, оскільки на великих форматах зображень наявний дисбаланс якості вихідних знімків різних класів нівелюється; зазначений ефект дозволяє розглядати таку зміну розмірів вхідних знімків як варіант аргументації даних у датасеті;
- заслуговує на увагу поєднання на вході звужуваного сегменту нейромережі класифікації не тільки симетрично збільшеного зображення, а й його варіантів, отриманих на основі узагальненого проникаючого добутку матриць з розширенням зображень по рядках пікселів (по горизонталі) та по стовпцях (по вертикалі); у цьому контексті можливе також поєднання кількох різних варіантів симетричних розширень зображень за принципом побудови пірамідального сегменту нейромережі PSPNet;
- запропоновані в рамках розвиненого тензорного матричного апарату нові варіанти операцій згорток та макспулінгів у поєднанні з архітектурним удосконаленням розширюють арсенал розробників нейромереж та закладають підгрунтя для нових напрямів відповідних досліджень.

Подальші зусилля доцільно зосередити на вивченні можливостей використання ковзної версії проникаючого торцевого добутку, зокрема, для формалізації опису механізму уваги, ковзної фільтрації даних по багатошаровому або багатовимірному тензору. Зазначений ковзний режим може бути запроваджений і для процедури збільшення вибіркових фрагментів у кадрі зображень, наприклад, при вирішенні

завдання детектування та трекінгу об’єктів класифікації.

References

1. Slyusar, V. I. (1998) End matrixs products in radar applications. *Radioelectronics and Communications Systems*, 41(3).
2. Slyusar V. I. (1997) New operations of matrices product for applications of radars, in Proc. Direct and Inverse Problems of Electromagnetic and Acoustic Wave Theory (DIPED-97), Lviv, 73-74 (in Russian). doi: 10.1109/DIPED.1997.710918.
3. Slyusar, V.I. (1999). A Family of Face Products of Matrices and its Properties. *Cybernetics and systems analysis c/c of Kibernetika i sistemnyi analiz. Consultants bureau (USA)*, 3(35), 379–384. doi: 10.1007/BF02733426.
4. Slyusar, V. I. (2003) Generalized face-products of matrices in models of digital antenna arrays with nonidentical channels. *Radioelectronics and Communications Systems*, 46(10), 9 - 17.
5. Slyusar, V. I. (1999) Fisher Information Matrix for Models of Systems Based on Face-Splitting Matrix Products, Cybernetics and Systems Analysis c/c of Kibernetika i Sistemnyi Analiz, 35(4), 636-643. doi: 10.1007/BF02835859.
6. Thomas D. Ahle, Jakob Bæk Tejs Knudsen (2019) Almost Optimal Tensor Sketch. *Mathematics, Computer Science, ArXiv*.
7. Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Gregory S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian J. Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Józefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dan Mané, Rajat Monga, Sherry Moore, Derek Gordon Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul A. Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda B. Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng (2016) Tensorflow: Large-scale machine learning on heterogeneous distributed systems. CoRR, abs/1603.04467, http://arxiv.org/abs/1603.04467.
8. Tensorflow, how to multiply a 2D tensor (matrix) by corresponding elements in a 1D vector (2017), https://stackoverflow.com/questions/47817135/tensorflow-how-to-multiply-a-2d-tensor-matrix-by-corresponding-elements-in-a.
9. Ha D., Dai A.M., Le Q.V. HyperNetworks (2017) *The International Conference on Learning Representations (ICLR*), Toulon, https://arxiv.org/abs/1609.09106.
10. LeCun, Y., Bottou, L., Bengio, Y. and Haffner, P. (1998) Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11), 2278- 2324.
11. Slyusar V.I. The tensor-matrix version of LeNet5 (2021) *IVth International scientific-practical conference dedicated to the 50th anniversary of the Department of Information Systems and Technologies «Integration Information Systems and Intelligent Technologies in the Conditions of Information Society Transformation»,* Poltava: Poltava State Agrarian University, 114 - 119. doi: 10.32782/978-966-289-562-9.
12. Krizhevsky, Alex; Sutskever, Ilya; Hinton, Geoffrey E. (2017). ImageNet classification with deep convolutional neural networks, *Communications of the ACM*. 60 (6): 84–90. doi:10.1145/3065386.
13. Vadym Slyusar, Mykhailo Protsenko, Anton Chernukha, Pavlo Kovalov, Pavlo Borodych, Serhii Shevchenko, Oleksandr Chernikov, Serhii Vazhynskyi, Oleg Bogatov, Kirill Khrustalev. Improvement of the object recognition model on aerophotos using deep conventional neural network.// Eastern-European Journal of Enterprise Technologies. - 2021, Vol. 5, No. 2 (113). Pp. 6 – 21. DOI: 10.15587/1729-4061.2021.243094.
14. H. Qassim, A. Verma and D. Feinzimer (2018), Compressed residual-VGG16 CNN model for big data places image recognition, *Computing and Communication Workshop and Conference (CCWC) 2018 IEEE 8th Annual*, 169-175.
15. G. Huang, Z. Liu, L. Van Der Maaten and K. Weinberger (2017), Densely Connected Convolutional Networks, in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA, 2261-2269. doi: 10.1109/CVPR.2017.243
16. Howard, A.G., Zhu, M., Chen, B., et al. (2017) MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications.
17. Sandler, M., Howard, A., Zhu, M., et al. (2018) Mobilenetv2: Inverted Residuals and Linear Bottlenecks. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, 18-23 June 2018, 4510-4520. https://doi.org/10.1109/CVPR.2018.00474.
18. Howard, A., Sandler, M., Chu, G., et al. (2019) Searching for Mobilenetv3. Proceedings of the IEEE International Conference on Computer Vision, Seoul, 27 October-2 November 2019, 1314-1324. https://doi.org/10.1109/ICCV.2019.00140.
19. Zoph, B., Vasudevan, V., Shlens, J., & Le, Q. V. (2018). Learning Transferable Architectures for Scalable Image Recognition, https://arxiv.org/abs/1707.07012.
20. Olaf Ronneberger, Philipp Fischer, and Thomas Brox (2015) U-Net: Convolutional Networks for Biomedical Image Segmentation, https://arxiv.org/pdf/1505.04597.pdf .
21. Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, Yoshua Bengio (2014) Generative Adversarial Nets, In Advances in Neural Information Processing Systems (NIPS), 2672–2680.
22. Slyusar V.I., Sliusar I.I. (2021) Lions of Neural Networks Zoo, *NeyromerezhnI tehnologIyi ta yih zastosuvannya NMTIZ-2021: zbirnik naukovyh prats XX Mizhnarodnoyi naukovoyi konferentsiyi «Neyromerezhny tehnologii ta yih zastosuvannya NMTIZ-2021»,* Kramatorsk: DDMA, 129 -133, doi: 10.13140/RG.2.2.17187.58405.
23. A. Krizhevsky (2009). Learning multiple layers of features from tiny images. Master’s thesis, Department of Computer Science, University of Toronto.
24. J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. (2009) ImageNet: A Large-Scale Hierarchical Image Database. *In CVPR09*.

25. Md Jahidul Islam, Peigen Luoy and Junaed Sattar (2020), Simultaneous Enhancement and Super-Resolution of Underwater Imagery for Improved Visual Perception, *Robotics: Science and Systems 2020 Corvalis, Oregon, USA, July 12-16, 2020* .
26. Fjodor Van Veen. The neural network zoo. - https://www.asimovinstitute.org/neural-networkzoo/.
27. Slyusar V.I. Multimodal quasi-fractal neural networks (2021) *NeyromerezhnI tehnologIyi ta yih zastosuvannya NMTIZ-2021: zbirnik naukovyh prats XX Mizhnarodnoyi naukovoyi konferentsiyi «Neyromerezhny tehnologii ta yih zastosuvannya NMTIZ-2021»*, Kramatorsk: DDMA, 134 -137.
28. Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, Jiaya Jia (2016) Pyramid Scene Parsing Network, https://arxiv.org/abs/1612.01105.

Література

1. Slyusar, V. I. (1998) End matrixs products in radar applications. *Radioelectronics and Communications Systems*, 41(3).
2. Слюсар В.И. (1997) Новые операции умножения матриц в радиолокационных приложениях. *Прямі та обернені задачі теорії електромагнітних та акустичних хвиль (DIPED-97).* - Львов, 73-74. doi: 10.1109/DIPED.1997.710918.
3. Slyusar, V. (1999). A Family of Face Products of Matrices and its Properties. *Cybernetics and systems analysis c/c of Kibernetika i sistemnyi analiz. Consultants bureau (USA)*, 3(35), 379–384. doi: 10.1007/BF02733426.
4. Slyusar, V. I. (2003) Generalized face-products of matrices in models of digital antenna arrays with nonidentical channels. *Radioelectronics and Communications Systems*, 46(10), 9 - 17.
5. Слюсар В.И. (1999) Информационная матрица Фишера для моделей систем, базирующихся на торцевых произведениях матриц. *Кибернетика и системный анализ,* 35(4), 636 - 643. doi: 10.1007/BF02835859.
6. Thomas D. Ahle, Jakob Bæk Tejs Knudsen (2019) Almost Optimal Tensor Sketch. *Mathematics, Computer Science, ArXiv*.
7. Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Gregory S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian J. Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Józefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dan Mané, Rajat Monga, Sherry Moore, Derek Gordon Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul A. Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda B. Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng (2016) Tensorflow: Large-scale machine learning on heterogeneous distributed systems. CoRR, abs/1603.04467, http://arxiv.org/abs/1603.04467.
8. Tensorflow, how to multiply a 2D tensor (matrix) by corresponding elements in a 1D vector (2017), https://stackoverflow.com/questions/47817135/tensorflow-how-to-multiply-a-2d-tensor-matrix-by-corresponding-elements-in-a.
9. Ha D., Dai A.M., Le Q.V. HyperNetworks (2017) *The International Conference on Learning Representations (ICLR*), Toulon, https://arxiv.org/abs/1609.09106.
10. LeCun, Y., Bottou, L., Bengio, Y. and Haffner, P. (1998) Gradient-based learning applied to document recognition. *Proceedings of the IEEE,* 86(11), 2278- 2324.
11. Слюсар В.І. Тензорно-матричная версия LeNet5 (2021) IV Міжнародна науково-практична конференція «Інтеграція інформаційних систем і інтелектуальних технологій в умовах трансформації інформаційного суспільства», що присвячена 50-ій річниці кафедри інформаційних систем та технологій, 21-22 жовтня 2021 р., Полтава: Полтавський державний аграрний університет, 114 - 119. doi: 10.32782/978-966-289-562-9.
12. Krizhevsky, Alex; Sutskever, Ilya; Hinton, Geoffrey E. (2017). ImageNet classification with deep convolutional neural networks, Communications of the ACM. 60 (6): 84–90. doi:10.1145/3065386.
13. Vadym Slyusar, Mykhailo Protsenko, Anton Chernukha, Pavlo Kovalov, Pavlo Borodych, Serhii Shevchenko, Oleksandr Chernikov, Serhii Vazhynskyi, Oleg Bogatov, Kirill Khrustalev. Improvement of the object recognition model on aerophotos using deep conventional neural network.// Eastern-European Journal of Enterprise Technologies. - 2021, Vol. 5, No. 2 (113). Pp.6–21. DOI: 10.15587/1729-4061.2021.243094.
14. H. Qassim, A. Verma and D. Feinzimer (2018), Compressed residual-VGG16 CNN model for big data places image recognition, Computing and Communication Workshop and Conference (CCWC) 2018 IEEE 8th Annual, 169-175.
15. G. Huang, Z. Liu, L. Van Der Maaten and K. Weinberger (2017), Densely Connected Convolutional Networks, in 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 2261-2269. doi: 10.1109/CVPR.2017.243
16. Howard, A.G., Zhu, M., Chen, B., et al. (2017) MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications.
17. Sandler, M., Howard, A., Zhu, M., et al. (2018) Mobilenetv2: Inverted Residuals and Linear Bottlenecks. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, 18-23 June 2018, 4510-4520. doi.org/10.1109/CVPR.2018.00474.
18. Howard, A., Sandler, M., Chu, G., et al. (2019) Searching for Mobilenetv3. Proceedings of the IEEE International Conference on Computer Vision, Seoul, 27 October-2 November 2019, 1314-1324. doi.org/10.1109/ICCV.2019.00140.
19. Zoph, B., Vasudevan, V., Shlens, J., & Le, Q. V. (2018). Learning Transferable Architectures for Scalable Image Recognition, https://arxiv.org/abs/1707.07012.
20. Olaf Ronneberger, Philipp Fischer, and Thomas Brox (2015) U-Net: Convolutional Networks for Biomedical Image Segmentation, https://arxiv.org/pdf/1505.04597.pdf.
21. Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, Yoshua Bengio (2014)

Generative Adversarial Nets, In Advances in Neural Information Processing Systems (NIPS), 2672–2680.
22. Слюсар В.И., Слюсарь И.И. (2021) Львы зоопарка нейростей. *Нейромережні технології та їх застосування НМТіЗ-2021: збірник наукових праць XX Міжнародної наукової конференції «Нейромережні технології та їх застосування НМТіЗ-2021».* - Краматорськ: Донбаська державна машинобудівна академія.
23. A. Krizhevsky (2009). Learning multiple layers of features from tiny images. Master's thesis, Department of Computer Science, University of Toronto.
24. J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. (2009) ImageNet: A Large-Scale Hierarchical Image Database. *In CVPR09*.
25. Md Jahidul Islam, Peigen Luoy and Junaed Sattar (2020), Simultaneous Enhancement and Super-Resolution of Underwater Imagery for Improved Visual Perception, *Robotics: Science and Systems 2020 Corvalis, Oregon, USA, July 12-16, 2020*
26. Fjodor Van Veen. The neural network zoo. - https://www.asimovinstitute.org/neural-networkzoo/.
27. Слюсар В.И. Мультимодальные квази-фрактальные нейросети (2021) *Нейромережні технології та їх застосування НМТіЗ-2021: збірник наукових праць XX Міжнародної наукової конференції «Нейромережні технології та їх застосування НМТіЗ-2021»,* Краматорськ: Донбаська державна машинобудівна академія, 134 -137.
28. Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, Jiaya Jia (2016) Pyramid Scene Parsing Network, https://arxiv.org/abs/1612.01105.

Стаття надійшла до редакції 14.04.22
Після обробки 27.05.22